Qualitative Evaluation of BSDF Measurement Interpolation in *Radiance*

Greg Ward, Anyhere Software

Introduction

The PAB-Opto pgII measurement device is capable of recording variable- and high-resolution reflection and transmission data from various materials. However, time considerations limit the number of incident directions that may be measured practically for a given material sample. For isotropic materials, only a single sweep of angles between 0 and 90° incidence is necessary, so 10° spacing that corresponds to the full Klems angle basis is often used. Even so, the density of reflected directions is generally much finer than 10°, so some degree of averaging is performed. For anisotropic materials, it is rare that all 145 Klems incident directions will be measured due to the time required to complete such a task (on the order of weeks). Although automation now makes this possible, the associated expense is beyond the budget of most projects, especially when multiple materials are involved.

Further to this, the Klems basis of 145 incident and 145 exiting directions is frequently inadequate for lighting simulation such as glare analysis, which requires a variable-resolution representation such as the tensor tree structure recently introduced in *Radiance*. This places further demands on the measurement side, needing as many as 16,000 incident and 16,000 exiting directions. In fact, the pgII can produce this order of exiting direction measurements, but the burden on incident directions is beyond this or any existing device. It is therefore necessary to interpolate incident directions to arrive at a complete description of a material's reflection and/or transmission.

Interpolation of BSDF data turns out to be an extremely challenging problem. Basic methods such as multidimensional linear interpolation fail with large errors because they do not account for the correlation between incident and exiting ray directions. More sophisticated interpolation techniques have been devised to account for certain correlations in particular material classes, but these do not generalize well to the types of exotic materials (films, microstructures, etc.) that interest us in the context of daylighting systems. We have therefore sought out and developed a more general scheme for incident angle interpolation that is *model-free* in the sense that it should fit the behavior of any material, provided that the 4-D BSDF is smooth relative to the density of captured values. (In other words, we require that the incident angles be sampled densely enough that the measured set exhibits local coherence.)

Interpolation Method Overview

Our BSDF interpolation method can be logically divided into three stages. In the first stage, we take each incident angle measurement and fit it to a radial-basis function, which in this case is a sum of Gaussian lobes corresponding to averaged reflection values and directions. In the second stage, we determine how these Gaussian lobes evolve between one incident angle and its neighbors by creating a mesh of incident directions on a hemisphere. In the third stage, we interpolate both incident and exiting directions to evaluate this 4-dimensional BSDF representation at the positions required by our preferred representation, either a tensor tree or a Klems matrix in our implementation.

Radial Basis Function Representation

To represent all possible outgoing directions for a given incident direction, we create a Radial Basis Function (RBF) to interpolate our sample measurements. The RBF is essentially a sum of Gaussian lobes of the form:

$$\sum_i a_i e^{-\theta_i^2 / 2r_i^2}$$

The a_i's and r_i's in the sum correspond to fitted coefficients and radii for each lobe. The θ_i's correspond to the angles computed between each lobe center and the exiting direction vector. A typical RBF will have several hundred lobes corresponding to a region of averaged measurements, as shown in Figure 1. The pink dots in this 3-D plot correspond to averaged and redistributed measurements from the pgII, shown in yellow. The green sheet is the actual sum of Gaussians, or RBF for this incident direction. Note how the density of measured points is highly anisotropic, and there are extra measurements (appearing as a spiral of points near the top) where the pgII has been instructed to increase sampling density near the mirror direction. This makes our job easier in the sense that we have data where we need it, but simultaneously more difficult as our RBF code must adapt to arbitrary changes in measurement point density over different parts of the distribution.

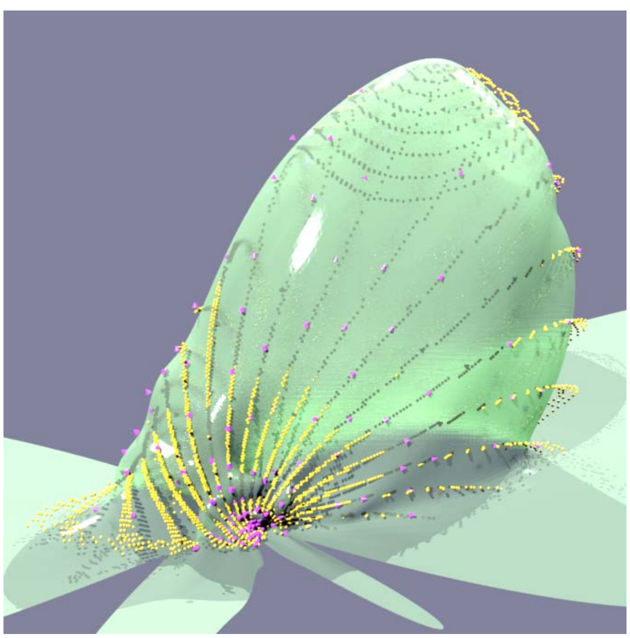

Figure 1. Radial basis function interpolating measured incident angle of an aluminum surface with a sawtooth profile. (Data courtesy Peter Apian-Bennewitz.)

Incident Angle Interpolation

Once we have a set of RBFs, one for each measured incident direction, we need some method for assembling these into a complete, 4-dimensional BSDF representation. As discussed earlier, any basic interpolation method will fail because outgoing directions change smoothly with a strong (unknown) correlation to incident direction. An elegant solution to this problem was described by Bonneel et al. in 2011 [1]. Working with the main author, we have adapted and extended his Langrangian mass transport method to handle BSDF interpolation as described in [4].

We start by arranging our measured incident directions into a Delaunay triangulation of the hemisphere. In many cases, we need measure only a quarter or a half of the hemisphere, as shown in Figure 2. This saves time when a system has known symmetry. Bilateral symmetry means only half of the incident hemisphere is measured. Quadrilateral symmetry implies we only need to cover a quarter of the incident hemisphere. Radial symmetry (i.e., an isotropic BSDF) requires only a 90° arc of incident angles be measured. Based on which incident directions have been measured, we deduce the appropriate symmetry and fill in the rest of the distribution accordingly.

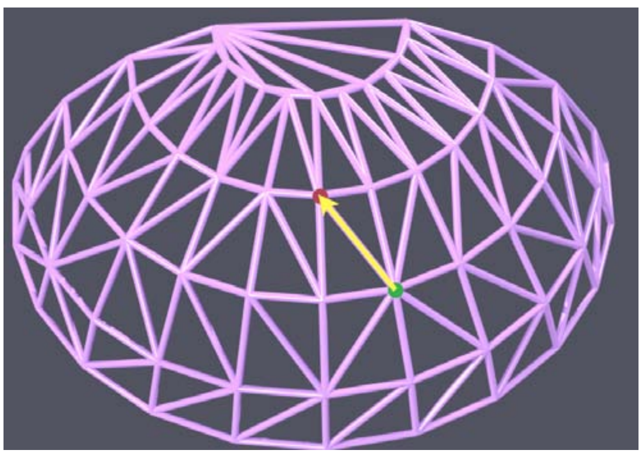

Figure 2. Delaunay interpolation of a half-hemisphere of incident directions.

Each vertex in Figure 2 corresponds to a measured incident direction, to which we have fitted a set of Gaussian lobes for a complete RBF representation of exiting directions. Along an edge between vertices, we use Lagrangian mass transport to transfer energy from the set of Gaussian lobes interpolating the first incident direction to the lobes for the second direction. Figure 2 shows one such "migration path" and Figure 3 diagrams the transport of "mass" between lobes in the neighboring RBF representations. This advection process interpolates the peak value, radius, and direction of each Gaussian lobe as it shifts from the output distribution at one incident direction to another. This behavior guarantees that BSDF peaks will move smoothly between interpolated directions.

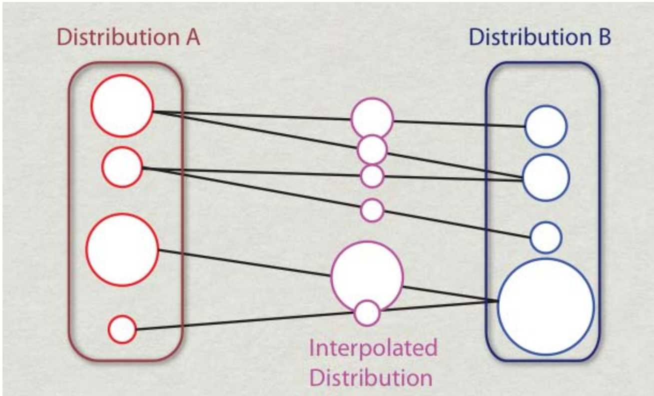

Figure 3. Advection process showing how Gaussian lobes in RBF migrate between distributions.

The mass in one lobe may be split as and redistributed to different destination lobes as it migrates, and this redistribution is characterized in a "migration matrix" with rows and columns corresponding to source and destination lobes, respectively. Minimizing the mass transport cost means that far-away lobes will have zero coefficients in our matrix, which is typically sparse as a result. Nevertheless, there will always be more lobes representing the in-between RBF than either the source or destination RBFs, which can lead to potentially long evaluation times for the interpolant. The problem gets worse when we interpolate arbitrary incident directions, which requires one edge migration matrix and a derived matrix corresponding to the interpolated point along an edge and a third vertex. In some cases, we might get an RBF with hundreds of thousands of lobes, which could make interpolation impractical. When this happens, we raise the threshold for "non-zero" migration coefficients until our lobe count is reduced to something more reasonable. (We found 15,000 lobes or fewer to be manageable.)

BSDF Characterization Pipeline

Mechanically, our software divides the interpolation problem into three slightly different stages for maximum efficiency. In the first stage, we take a set of pgII measurements corresponding to a particular pair of incident and exiting hemispheres, create an RBF for each incident direction, organize them into a Delaunay mesh, then compute migration matrices along each edge as described above. A material that reflects as well as transmits will have as many as four sets of RBF+matrix sets, which we call *Scattering Interpolation Representations*, or SIR files for short. Four SIR files would cover all combinations of front and back incident and exiting hemispheres. In the second stage, one or more SIR files are combined and interpolated into a preferred representation for simulation, either a Klems matrix or a tensor tree (XML) file. In the third stage, the preferred representation is applied in an analysis or simulation tool such as WINDOW6 or *Radiance*.

Breaking the interpolation problem up this way means we only need to compute the interpolant once, and the intermediate SIR files take the place of measurement data from that point forward. Multiple BSDF representations may be derived from one set of SIR files with reasonable efficiency, but the final XML file is better tailored for a particular simulation tool or method, and provides the most efficient calculation. These steps are highlighted in Table 1.

Table 1. BSDF calculation stages.

Calculation Stage	Tool(s)	Input	Output
Derive Interpolant	**pabopto2bsdf**	pgII measurements	Scattering interpolation representation (SIR)
Interpolate BSDF	**bsdf2klems** **bsdf2ttree**	SIR file(s) or procedural BSDFs	XML file
Simulation	WINDOW6 **rpict** **rtrace** **rcontrib**	XML file(s)	Combined BSDFs, Renderings, Annual calculations, etc.

Tensor Tree Representation

The tensor tree representation is particularly well-suited to rendering, as it provides higher resolution information near BSDF peaks while taking an acceptable amount of memory by reducing resolution in areas where little is going on. Figure 4 shows a false color plot of the transmitted intensity for a particular specular blinds system at one incident direction. Note how sparsely the tensor tree represents areas where the distribution is constant (or near zero). A similar subdivision happens on the incident side, so less than 10% of the full-resolution values are needed in the final XML file. At the same time, the tensor tree structure simplifies ray sampling, as described in [4].

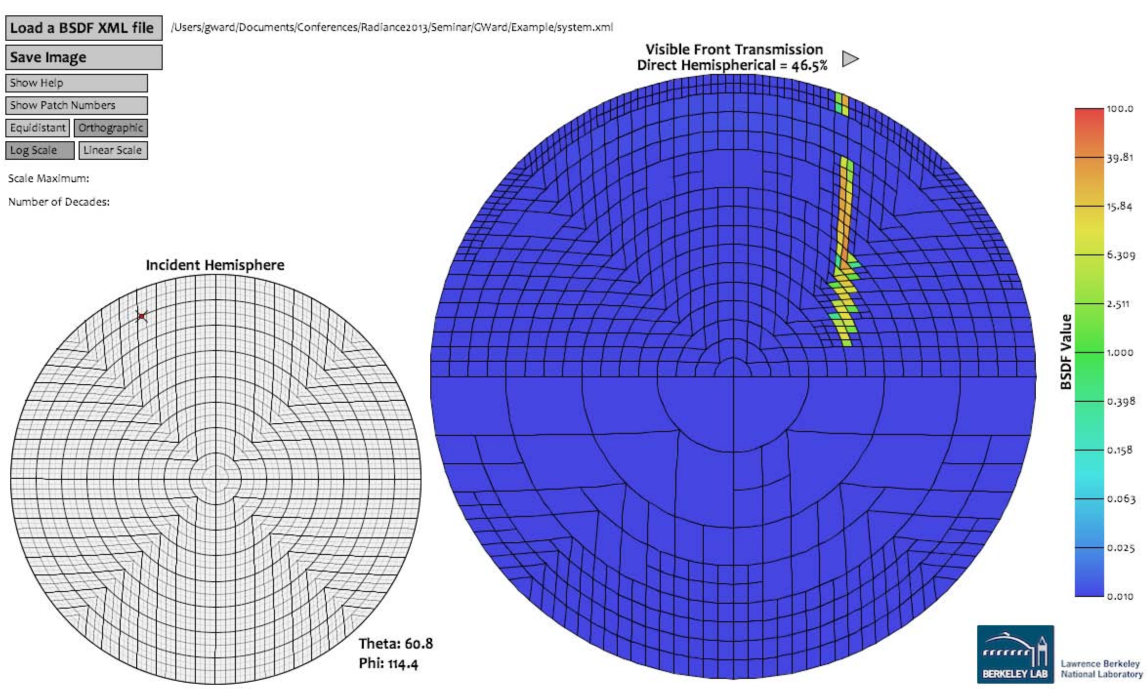

Figure 4. Plot of tensor tree representation. (BSDF Viewer by Andrew McNeil.)

Qualitative Evaluation of Interpolation

As described, we now have a complete system for taking pgII measurements and interpolating the results into a tensor tree representation for lighting simulation and rendering. The post-processing takes some time, but not nearly as long as the actual measurements, and need only be done once for each desired representation. We therefore consider this a practical method from the standpoint of a streamlined material characterization pipeline.

To evaluate the performance of our interpolation method, we need to devise a reasonable test case. It is neither required nor possible to determine the ground truth of a material's BSDF from a finite set of measurements, particularly when two or more dimensions (the incident directions) are sparsely sampled. On the other hand, the ground truth of a material would be very helpful in evaluating our interpolation method, because it would tell us where we were deviating from reality.

Although we cannot know the ground truth of any real material, it is relatively easy to *simulate* measurements of a mathematical BSDF model. We can even compare the relative merits of various sampling patterns and densities, which the pgII offers in abundance. We therefore apply the following strategy for evaluating our interpolation method using ground-truth BRDF models:

1) Select one of *Radiance*'s built-in material types and a set of associated parameters.
2) Render a suitable test scene that demonstrates the material's behavior.
3) Repeat this rendering using a Klems matrix or tensor tree in an XML file to show the effects of representation on the results.
4) Simulate physical material measurements by evaluating the BRDF at sample positions corresponding to a particular pgII sequence.
5) Apply our 3-stage interpolation method to reconstruct the BRDF as a Klems matrix or tensor tree representation.
6) Repeat the rendering from step (2) using the interpolated representation and compare the results to step (3).
7) Repeat as desired with different material types and parameters, scene configurations, interpolated representations, etc.

Test Scene and Materials

For our test scene, we followed the recommendations of previous researchers ([2] and [3]), who found that blob-like objects with convex and concave elements in natural surroundings were superior for differentiating material appearances. We created a blobby object and put it in an HDR forest environment, adding a single local light source to demonstrate highlight behavior. We animated our scene by rotating the object and orbiting the light source to show material behavior at all incident angles and look for unusual behavior in the interpolated BRDFs. (See complete animations at gaia.lbl.gov/people/andy/share/InterpValidAnim.)

Reference Renderings

The two reference renderings shown in Figures 5 and 6 use *Radiance*'s built-in isotropic and anisotropic BRDF models, respectively. All images should be compared to these ground truth starting points. The position of the object and light source is identical in the two images. The only difference is that the second rendering has narrower more intense highlights due to the anisotropy in the reflectance distribution. This provides a greater challenge to all stages of the BRDF characterization pipeline.

Figure 5. Isotropic material – reference rendering.

Figure 6. Anisotropic material – reference rendering.

Klems Matrix Representation

Figures 7 and 8 show the appearance of our two BRDF models when represented using a 145x145 Klems matrix. Our interpolation method was not used for this evaluation, which relied on the ground truth of the material model only. Any artifacts are a result of the low-density sampling of the Klems basis.

Figure 7. Klems representation of isotropic material model.

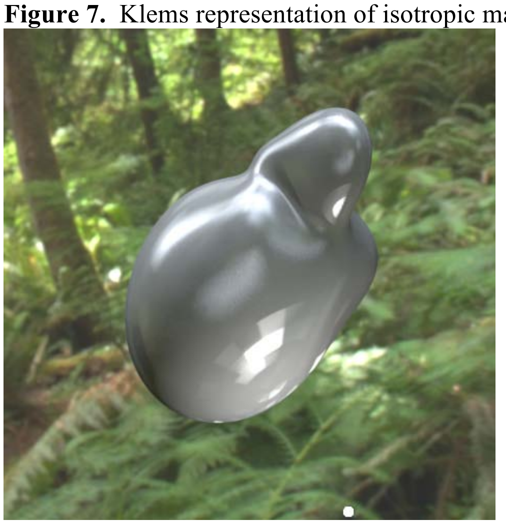

Figure 8. Klems representation of anisotropic material model.

Tensor Tree Representation

Figures 9 and 10 show the appearance of our two BRDF models when represented using the variable-resolution tensor tree. Our interpolation method was not used for this conversion, which relied on the ground truth of the material model only. Any artifacts are purely a result of the tensor tree representation.

Figure 9. Tensor tree representation of isotropic material model.

Figure 10. Tensor tree representation of anisotropic material model.

Interpolated Klems Representation

Figures 11 and 12 show an interpolated set of data points converted to a 145x145 Klems matrix representation. Most of the visible artifacts resemble those of the previous Figures 7 and 8, and are due to the resolution limitations of the Klems matrix itself.

Figure 11. Interpolated Klems representation of isotropic material model.

Figure 12. Interpolated Klems representation of anisotropic material model.

Interpolated Tensor Tree Representation

Figures 13 and 14 are rendered using tensor tree BRDFs interpolated from virtual measurements of the original (ground truth) functions. The resolution of these representations is not as fine as the previous tensor tree renderings in Figures 9 and 10. We believe this is mostly due to an overly coarse sampling pattern, which needs to be finer to resolve this function, particularly in the anisotropic case. We plan to investigate this further in our ongoing validation efforts.

Figure 13. Interpolated tensor tree representation of isotropic material model.

Figure 14. Interpolated tensor tree representation of anisotropic material model.

Discussion and Future Work

It is clear from our initial study that the sampling pattern makes an important difference to the results. While two of the sampling patterns we used produced acceptable results for the isotropic case using the tensor tree representation, one of our two anisotropic patterns did not perform very well at all.

Figure 15 shows the same anisotropic BRDF sampled using the anisotropic pattern shown for one angle of incidence in Figure 16. While this pattern has dense sampling in the mirror direction, the high-density region is most likely too small to contain the full highlight. Compare this pattern to the one shown in Figure 17, which we used to produce the BRDF interpolation shown in Figure 14 above. The wider highlight sampling area of the latter pattern clearly works better in this case, but we need to investigate further using other patterns to understand this issue.

Figure 15. Interpolated tensor tree using anisotropic pattern shown in Figure 16.

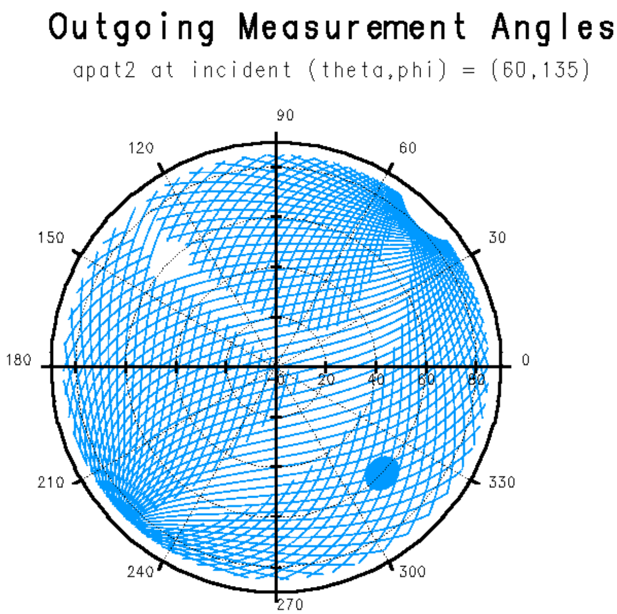

Figure 16. Sampling pattern that produced poor BRDF interpolation shown in Figure 15.

Once we understand the basic sensitivities of our interpolation method, we can optimize our pgII sampling patterns and characterize the local and global errors quantitatively. In particular, we need to consider the effect of BSDF accuracy on calculated results, such as annual simulations, glare analyses, and point (luminance, illuminance) evaluations.

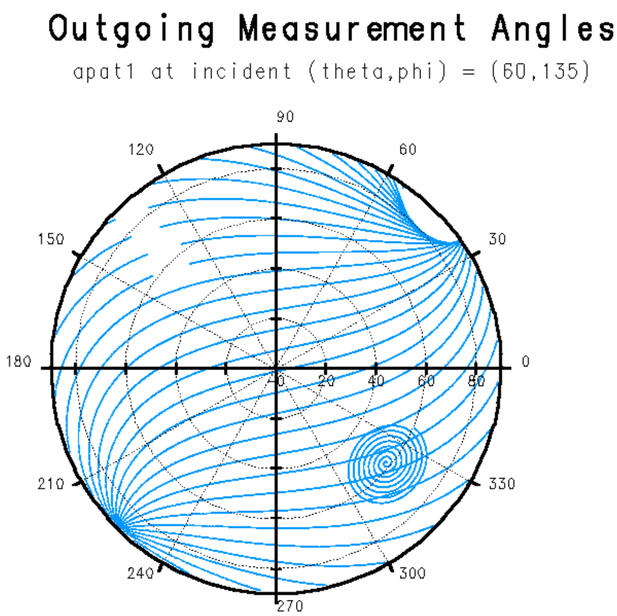

Figure 17. Broader highlight sampling pattern that produced the results shown earlier in Figure 14.

References

[1] Nicolas Bonneel, M. van de Panne, W. Heidrich. Displacement interpolation using Lagrangian mass transport. ACM Transactions on Graphics 30, 6. 2011.
[2] Roland W. Fleming, Ron O. Dror, and Edward H. Adelson. Real-world illumination and the perception of surface re- flectance properties. Journal of Vision, 3(5), 2003.
[3] Peter Vangorp, Jurgen Laurijssen, and Philip Dutre. The influence of shape on the perception of material reflectance. ACM Trans. Graph., 26, July 2007.
[4] Gregory Ward, Murat Kurt, Nicolas Bonneel. A Practical Framework for Sharing and Rendering Real-World Bidirectional Scattering Distribution Functions. LBNL-5954E. buildings.lbl.gov/publications/practical-framework-sharing-and-rende Oct. 2012.